



Red Hat build of Cryostat 2

Red Hat build of Cryostat 2.4 のリリースノート

Red Hat build of Cryostat 2 Red Hat build of Cryostat 2.4 のリリースノート

法律上の通知

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

Red Hat build of Cryostat 2.4 のリリースノート には、Cryostat 2.4 の新機能の概要と、潜在的な既知の問題と考えられる回避策のリストが記載されています。

目次

はじめに	3
多様性を受け入れるオープンソースの強化	4
第1章 CRYOSTAT のサポートポリシー	5
第2章 新機能	6
ARM64 アーキテクチャーのサポート	6
MBean カスタムトリガーに基づく動的な JFR レコーディング	6
JFR 書き込み操作に対するエージェント HTTP API のサポート	6
デュアル HTTP および JMX 検出の登録	6
依存関係のない標準エージェント JAR ファイル	6
エージェント JAR ファイルディストリビューションの選択	7
エージェントが登録できるサーバーホストまたは IP アドレス範囲のデフォルトの制限	7
第3章 機能拡張	9
Cryostat エージェントを表示するためのトポロジビューの機能拡張	9
第4章 修正された問題	11
遅延接続ターゲットでのルールアクティベーションの失敗	11
検出プラグインの登録の失敗	11
エージェントインスタンスが利用できないため、サーバーの起動に失敗する	11
cryostat-reports および jfr-datasource によるコンテナ調整プロパティがオーバーライドされてしまう	11
自動ルールを使用してフライトレコーディングを再開するときに JFR データが切り捨てられる	11
CPU 制限が小さいためにコンテナの起動が失敗する	12
第5章 既知の問題	13
Cryostat エージェントはレコーディング開始リクエストで ALL イベントテンプレートに対応できない	13
Red Hat Insights のインテグレーションが ARM64 アーキテクチャーで失敗する	13
第6章 このリリースに関連するアドバイザリー	15

はじめに

Red Hat build of Cryostat は、JDK Flight Recorder (JFR) のコンテナネイティブ実装です。これを使用すると、OpenShift Container Platform クラスターで実行されるワークロードで Java 仮想マシン (JVM) のパフォーマンスを安全にモニターできます。Cryostat 2.4 を使用すると、Web コンソールまたは HTTP API を使用して、コンテナ化されたアプリケーション内の JVM の JFR データを起動、停止、取得、アーカイブ、インポート、およびエクスポートできます。

ユースケースに応じて、Cryostat が提供するビルトインツールを使用して、Red Hat OpenShift クラスターに直接レコーディングを保存して分析したり、外部のモニタリングアプリケーションにレコーディングをエクスポートして、レコーディングしたデータをより詳細に分析したりできます。



重要

Red Hat build of Cryostat は、テクノロジープレビュー機能のみです。テクノロジープレビュー機能は、Red Hat 製品のサービスレベルアグリーメント (SLA) の対象外であり、機能的に完全ではないことがあります。Red Hat は、実稼働環境でこれらを使用することを推奨していません。テクノロジープレビュー機能は、最新の製品機能をいち早く提供して、開発段階で機能のテストを行いフィードバックを提供していただくことを目的としています。

Red Hat のテクノロジープレビュー機能のサポート範囲に関する詳細は、[テクノロジープレビュー機能のサポート範囲](#) を参照してください。

多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、今後の複数のリリースで段階的に用語の置き換えを実施して参ります。詳細は、[Red Hat CTO である Chris Wright のメッセージ](#) をご覧ください。

第1章 CRYOSTAT のサポートポリシー

Red Hat は、Cryostat のメジャーバージョンを最低 6 カ月間サポートします。Red Hat は、製品が Red Hat カスタマーポータルでリリースされる時期に基づいてこの数値を定めています。

Cryostat は、x86_64 または ARM64 アーキテクチャーで実行される Red Hat OpenShift Container Platform 4.11 以降のバージョンにインストールしてデプロイできます。

関連情報

- Cryostat のライフサイクルポリシーの詳細は、Red Hat OpenShift Container Platform ライフサイクルポリシーの Web ページにある [Red Hat build of Cryostat](#) を参照してください。

第2章 新機能

Cryostat 2.4 では、Cryostat 製品の使用を強化する新機能が導入されています。

ARM64 アーキテクチャーのサポート

Cryostat 2.4 では、ARM64 (aarch64) アーキテクチャーで実行される Red Hat OpenShift Container Platform 4.11 以降のバージョンで Cryostat の使用のサポートを開始しています。

MBean カスタムトリガーに基づく動的な JFR レコーディング

Cryostat 2.4 では、Cryostat エージェントを有効にして、MBean のカスタムトリガーに基づいて JFR レコーディングを動的に開始できます。

カスタムトリガー条件は、ランタイム、メモリー、スレッド、オペレーティングシステムのメトリックの範囲に対応できる MBean カウンターに基づいています。JFR レコーディングのカスタムトリガー条件の一部として1つ以上の MBean カウンタータイプを含めることができます。トリガー条件の一部として期間を指定することもできます。つまり、条件が満たされるまで、条件値は指定された期間存続する必要があります。各カスタムトリガー定義には、Cryostat エージェントがレコーディングに使用できるイベントテンプレートも含まれている必要があります。

Cryostat エージェントは、指定された MBean カウンターの値を継続的にリッスンするスマートトリガーをサポートします。指定されたカウンターの現在の値が、指定された期間のカスタムトリガーの設定値と一致すると、トリガーが発生します。トリガーが発生すると、Cryostat エージェントは指定されたイベントテンプレートに基づいて、その時点で JFR レコーディングを動的に開始します。



注記

このレコーディングに関連するカスタムトリガー条件が満たされない場合、JFR レコーディングは動的に開始されません。

JFR 書き込み操作に対するエージェント HTTP API のサポート

Cryostat 2.4 エージェントは、Cryostat サーバーがアプリケーションの JMX ポートの代わりに使用できる HTTP API を提供します。デプロイするワークロードアプリケーションに適切に設定された Cryostat 2.4 エージェントを割り当てることで、ターゲットアプリケーションが JMX ポートを公開する必要なく、完全な Cryostat 機能セットを使用できます。

今回の機能拡張により、Cryostat エージェントはオンデマンドリクエストに対応し、エージェント HTTP 接続があるターゲット JVM の JFR レコーディングを開始、停止、および削除できます。これは、Cryostat エージェントが限られた JFR 操作のみをサポートする読み取り専用 HTTP API を提供していた以前のリリースの動作に取って代わります。



注記

デフォルトでは、Cryostat エージェントはサーバーからのオンデマンド要求に基づいて JFR レコーディングを開始および停止できません。Cryostat エージェントがこれらのタイプのリクエストを処理するようにする場合は、ターゲットアプリケーションの設定で **cryostat.agent.api.writes-enabled** プロパティを **true** に設定する必要があります。

デュアル HTTP および JMX 検出の登録

Cryostat 2.4 エージェントがターゲットアプリケーション上でも JMX が設定されていることを検出すると、エージェントはエージェント HTTP API 定義と JMX URL 定義の両方を使用して自身を Cryostat サーバーに公開します。この場合は、任意の設定オプションを使用できます。

依存関係のない標準エージェント JAR ファイル

Cryostat 2.4 では、依存関係のないエージェントコードを含む標準 JAR ファイルを使用して Cryostat エージェントをデプロイすることができます。詳細は、[エージェント JAR ファイルディストリビューションの選択](#) を参照してください。

エージェント JAR ファイルディストリビューションの選択

Cryostat 2.4 は、Cryostat エージェントの JAR ファイルとして 2 つのバリエーションを配布しています。設定要件に応じて、次のタイプのエージェント JAR ファイルのいずれかを使用できます。

- 自己完結型で、エージェントコードとそのすべての依存関係が含まれるオールインワンの "shaded" JAR ファイル
このファイルでは、別途追加する必要があるのはエージェント JAR ファイルを 1 つだけであるため、既存のアプリケーションに組み込むのに最も便利な形式の Cryostat エージェントが提供されます。
- 依存関係のないエージェントコードを含む標準 JAR ファイル
これは、エージェントとワークロードアプリケーションの間に依存関係の競合が存在することがわかっており、依存関係の競合を解決するために独自のストラテジーを適用する予定がある場合に便利です。

これは、オールインワンの "shaded" JAR ファイルとしてエージェントのディストリビューションを 1 つだけ提供していた以前のリリースの動作を置き換えます。

エージェントが登録できるサーバーホストまたは IP アドレス範囲のデフォルトの制限
Cryostat 2.4 では、Cryostat エージェントが自身をアドバタイズする Cryostat サーバーのリソースの場所 (URL) に制限を設定できる新しい **cryostat.agent.baseuri-range** システムプロパティーが導入されています。Cryostat エージェントは **cryostat.agent.baseuri-range** プロパティーを使用して、**cryostat.agent.baseuri** プロパティーで指定した URL が、受け入れ可能なホスト名または IP アドレス内にあるかどうかを確認できます。

この機能は、誤って設定すると、エージェントが不正なサービスまたは管理外のサービスにアドバタイズする可能性があるため、ユーザーがサーバーのベース URI を誤って設定することを防ぐのに役立ちます。サーバーの URL の設定が間違っていると、Cryostat エージェントインスタンスからクラスター外のサービスへのデータ漏洩、またはオープンインターネット経由でのデータのルーティングにつながる可能性があります。

適用するセキュリティ制限のレベルに応じて、**cryostat.agent.baseuri-range** プロパティーを次の値のいずれかに設定できます。

- **loopback**
これにより、最高レベルの制約が提供されます。エージェントは、サーバー URL に **localhost** や **127.0.0.1** などのアドレスがある場合にのみサーバーに接続します。
- **link_local**
これにより、2 番目に高いレベルの制限が提供されます。エージェントは、サーバー URL のアドレスがリンクローカル範囲内のアドレスであるか、すでに最高レベルの制限内にあるアドレスが指定されている場合にのみサーバーに接続します。
- **site_local**
これにより、3 番目の高い制限レベルが提供されます。エージェントは、サーバー URL に IPv6 site-local または IPv4 のプライベートアドレス、またはより高いレベルの制限内にアドレスが指定されている場合にのみサーバーに接続します。
- **dns_local**
これにより、最も高い制限レベルが提供されます。エージェントは、サーバー URL に **.local** の接尾辞または **.localhost** の接尾辞が付いたアドレス、またはより高いレベルの制限内にあるアドレスがある場合にのみサーバーに接続します。



注記

これはデフォルト設定です。Cryostat サーバーは通常、**<https://cryostat.my-namespace.svc.cluster.local>** などの内部 OpenShift Service アドレスによって参照されるため、このレベルの制限は Red Hat OpenShift Container Platform での Cryostat サーバーおよびエージェントのデプロイメントに適しています。

- **public**

これにより制限が削除されます。この設定は、エージェントが設定された任意のホストまたは IP アドレスに接続できた以前のリリースの動作と同じにします。

第3章 機能拡張

Cryostat 2.4 には、Cryostat 2.3 の機能に基づいて構築された機能拡張が含まれています。

Cryostat エージェントを表示するためのトポロジービューの機能拡張

Cryostat Web コンソールのトポロジービューでは、Cryostat エージェントが Cryostat アイコンとともに表示されるようになりました。これは、Cryostat エージェントを OpenJDK Duke アイコンとともに表示される JMX ターゲットと区別するのに役立ちます。

トポロジービューとダッシュボードビューには追加のターゲット詳細が表示されます

Cryostat Web コンソールのトポロジービューとダッシュボードビューには、ターゲット JVM に関する次の追加タイプの情報が表示されるようになりました。

- オペレーティングシステム名。
- 物理メモリーの合計
- 合計スワップ領域
- クラスパス
- ライブラリーパス
- 入力引数
- システムプロパティー



注記

この情報は以下のいずれかの方法で表示できます。

- トポロジーパネルで、ツイスト (>) アイコンをクリックしてターゲットエンドポイントまでデプロイメントし、このエンドポイントの **Details** タブをクリックします。
- ダッシュボードパネルで、ターゲット JVM の詳細カードをダッシュボードに追加します。

JFR 記録の自動分析の機能拡張

Cryostat 2.4 Web コンソールでは、JFR レコーディングの自動分析情報は自動分析ダッシュボードカードのコンポーネントを再利用します。これは、自動分析データが提供する情報セットが限定的な以前のリリースの動作に取って代わります。



注記

JFR レコーディングの自動分析情報を表示するには、レコーディングのリストが含まれる任意のパネルを開き、関心のあるレコーディングの横にあるツイスト (>) アイコンをクリックします。

JFR レコーディングを再起動するためのパラメーターの強化

JFR フライトレコーディングを作成し、同じターゲット上に同じ名前の別の記録がすでに存在する場合、Cryostat サーバーは通常、レコーディングの作成リクエストを拒否します。クライアントは、要求パラメーターを指定してレコーディングを再開することで、この動作を変更できます。

以前のリリースでは、クライアントは記録を作成するリクエストに **再起動** パラメーターを含めること

ができました。**restart** パラメーターには **true** または **false** の値を指定でき、デフォルトでは **false** に設定されていました。クライアントリクエストに **restart=false** 設定が含まれる場合、同じ名前の別のレコーディングがすでに存在する場合、Cryostat サーバーはリクエストを拒否します。クライアント要求に **restart=true** 設定が含まれている場合、Cryostat サーバーは同じ名前の既存の記録を停止、削除して、再作成します。

Cryostat 2.4 では、以前のリリースで使用できた **restart** パラメーターよりも新しい **replace** パラメーターが導入されました。**replace** パラメーターには、**always**、**none**、または **stop** の値を使用できます。このような場合、**replace=always** の設定は古い **restart=true** の動作と一致し、**replace=never** の設定は古い **restart=false** の動作と一致します。**replace=stopped** 設定は、既存のレコーディングが **STOPPED** 状態にある場合にのみ、同じ名前でのレコーディングを再作成するようにサーバーに指示することで、3 番目のタイプの動作を提供します。それ以外の場合、クライアント要求に **replace=stopped** 設定が含まれているが、同じ名前の既存のレコーディングが **RUNNING** などの別の状態にある場合、サーバーは要求を拒否します。

この拡張機能では、自動化ルールで JFR レコーディングを再作成できるようにし、状態が **STOPPED** ではないレコーディングのアーカイブデータが失われる可能性がある状況を回避できます。

エラーおよび警告メッセージの強化

Cryostat 2.4 には、さまざまなエラーおよび警告メッセージの拡張が含まれています。

第4章 修正された問題

Cryostat のリリースには、Cryostat の以前のリリースで特定された問題の修正が含まれている場合があります。問題の説明とその後の修正については、修正された各問題の注記を確認してください。

Cryostat 2.4 リリースで、以下の問題が修正されました。

遅延接続ターゲットでのルールアクティベーションの失敗

Cryostat 2.4 より前のバージョンでは、Cryostat は検出された JVM ターゲットで自動ルールをトリガーできない場合があります。この問題は、コンテナ内の JVM が受信 JMX リクエストを受け入れる準備ができる前に、検出メカニズム (OpenShift エンドポイントのクエリーなど) により新しいターゲットコンテナに関する通知を Cryostat に送信した場合に発生する可能性があります。

Cryostat 2.4 は、以前は接続できなかったターゲット JVM の再チェックを実行し、設定された期間、定期的に頻繁にルールのトリガーを再試行することで、この問題を解決します。今回の修正は、Cryostat と JVM の間の接続の問題をより迅速に解決するのに役立ちます。この修正は、ルールの有効化と無効化を切り替えるときに、ルールがトリガーされる一貫性のない動作や予測不可能な動作を回避するのにも役立ちます。

検出プラグインの登録の失敗

Cryostat 2.4 より前では、Cryostat エージェントのインスタンスが Cryostat サーバーに登録できない場合があります。登録プロトコルには、このような問題を特定して登録をリセットできるように、定義された動作がありますが、このような定義済み動作は、他の状況でも失敗する可能性があります。これらの問題が両方発生した場合、Cryostat サーバーには、サーバーが有効であるとみなした無効なエージェント登録レコードが存在します。この状況では、エージェントは、以前の登録が失敗したという想定に基づいて、さらに登録を試行する可能性があります。ただし、サーバーはその後、エージェントがすでに登録されているという想定に基づいて、エージェントの再登録の試みを拒否します。これにより、エージェントとサーバーは相互に認識したり、登録ステータスをリセットしたりすることはできません。

Cryostat 2.4 はこの問題を解決し、より信頼性の高いエージェント検出登録を提供します。

エージェントインスタンスが利用できないため、サーバーの起動に失敗する

Cryostat 2.4 より前では、Cryostat サーバーをシャットダウンして再起動すると、サーバーが正常に再起動できない場合があります。この問題は、有効なエージェント登録レコードが含まれるサーバーをシャットダウンし、エージェントを再起動せずにこれらのエージェントインスタンスをシャットダウンした場合に発生していました。このような場合、サーバーの起動時に、サーバーは既存のエージェント登録レコードを確認して、これらのエージェントインスタンスがまだ存在すると想定します。しかし、エージェントが利用できなくなったため、サーバーは起動に失敗しました。

Cryostat 2.4 ではこの問題が解決されています。このリリースでは、サーバーの起動時に、利用できなくなったエージェントインスタンスにエージェントの登録レコードがまだ存在する場合、サーバーはこれらの登録レコードを削除し、サーバーの起動が通常どおり続行されます。

cryostat-reports および jfr-datasource によるコンテナ調整プロパティがオーバーライドされてしまう

cryostat-reports および **jfr-datasource** コンテナはどちらも OpenJDK UBI ランタイムイメージを使用します。Cryostat 2.4 より前では、これらのコンテナは、基本イメージのエントリーポイントスクリプトが実行するさまざまなコンテナ調整プロパティをオーバーライドするように誤って設定されていました。

Cryostat 2.4 では、**cryostat-reports** と **jfr-datasource** コンテナがコンテナ調整プロパティをオーバーライドしなくなり、これらのパラメーターが有効になりました。

自動ルールを使用してフライトレコーディングを再開するときに JFR データが切り捨てられる
Cryostat 2.4 より前のバージョンでは、クライアントが自動化ルールを使用して JFR レコーディングを

作成する要求を送信すると、既存のレコーディングの JFR データが予期せず切り捨てられる可能性があります。これは、クライアント要求に **restart=true** 設定が含まれているが、同じ名前の既存のレコーディングがまだ **STOPPED** 状態ではなかった場合に発生する可能性があります。この状況では、サーバーがレコーディングを自動的に停止、削除、再作成したため、アーカイブされていない録画データが失われました。

Cryostat 2.4 では、以前のリリースで使用できた **restart** パラメーターに代わる **replace** パラメーターを導入することで、この問題を解決しています。JFR レコーディングを作成する自動ルールを定義するときに、クライアント要求に **replace=stopped** 設定を含めることができるようになりました。この設定は、既存のレコーディングが **STOPPED** 状態にある場合にのみ、JFR レコーディングを再開するようにサーバーに指示します。既存のレコーディングが **RUNNING** などの別の状態にある場合、サーバーはリクエストを拒否します。

CPU 制限が小さいためにコンテナの起動が失敗する

Cryostat 2.4 より前は、CPU の制限が小さく、Cryostat がデプロイされている場合、コンテナが起動に失敗する可能性があります。

Cryostat 2.4 では、Cryostat が CPU 制限が小さい状態でデプロイされている場合でもコンテナを起動できるようにすることで、この問題を解決しています。

第5章 既知の問題

Cryostat のリリースには、Red Hat が認識し、後の製品開発段階で修正される可能性がある問題が含まれている場合があります。既知の各問題の説明と解決策を確認してください。

Cryostat エージェントはレコーディング開始リクエストで ALL イベントテンプレートに対応できない

説明

Cryostat エージェントを使用しているターゲット JVM の JFR レコーディングを作成するときに、ALL イベントテンプレートを選択すると、Cryostat エージェントは HTTP 400 エラーを返します。

回避策

要件に応じて、次のいずれかを実行できます。

- ALL イベントテンプレートを使用する場合は、ターゲット JVM との JMX 接続を使用します。
- ターゲット JVM で Cryostat エージェント接続を使用する場合は、Profiling などの別のイベントテンプレートを使用します。

Red Hat Insights のインテグレーションが ARM64 アーキテクチャーで失敗する

説明

Cryostat Operator の namespace では、名前が **Insights-proxy** で始まる Pod が **ImagePullBackOff** のエラー状態になります。これは、この Pod に使用される **APICast** コンテナが ARM64 アーキテクチャー用に構築されていないためです。ただし、Cryostat と Cryostat エージェントを使用するすべてのアプリケーションは引き続き正常に機能するはずですが。

回避策

Red Hat Insights のインテグレーションを無効にするには、以下を実行します。

1. **Operators > Installed Operators** に移動します。
2. Red Hat build of Cryostat Operator を選択します。
3. **Subscription** タブを選択します。
4. **Actions** ドロップダウンメニューから、**Edit Subscription** を選択します。
5. **INSIGHTS_ENABLED** 環境変数を **Subscription** オブジェクトに追加します。

```
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: cryostat-operator
  namespace: openshift-operators
spec:
  config:
    env:
      - name: INSIGHTS_ENABLED
        value: "false"
  channel: stable
  installPlanApproval: Automatic
  name: cryostat-operator
```

```
source: my-operator-catalog
sourceNamespace: openshift-marketplace
startingCSV: cryostat-operator.v2.4.0
```

6. **Save** をクリックします。

この回避策は、Cryostat Operator での Red Hat Insights インテグレーションを無効にし、**insights-proxy** デプロイメントを削除します。

第6章 このリリースに関連するアドバイザリー

Cryostat 2.4 リリースに含まれるバグ修正と CVE 修正を文書化するために、次のアドバイザリーが発行されています。

- [RHSA-2023:7669](#)

改訂日時: 2024-01-02